Diploma Thesis

A GUI-based Interaction Concept for Efficient Slide Layout

Volker C. Schöch
vschoech@inf.fu-berlin.de

February 2003

Freie Universität Berlin
Institut für Informatik
Takustr. 9
14195 Berlin
Germany

Reviewers: Prof. Dr. Raúl Rojas
Prof. Dr. Klaus-Peter Löhr

Advisors: Dr. Markus Hannebauer
Felix von Hundelshausen

Abstract

A range of research systems have been developed for layout automation, but none of them suitable for business slide creation. There are, also, many efficient user interface techniques described in the literature, but until today none of them have been employed in popular presentation software. In this thesis, a field study was conducted that shows which problems arise when Microsoft PowerPoint is being used in a professional context. Based on the study, this research presents a new interaction concept for the efficient layout of business slides.

The *smart grid* is a technique that is used to numerically encode spatial constraints and to arrange *smart elements* on a slide accordingly. An intuitive interaction concept was developed that allows the efficient specification of smart grid constraints. The concept was implemented in a prototype, using efficient algorithms to achieve interactive response times. A dynamic programming algorithm is used to compute intelligent layout suggestions.

The prototype runs as part of a commercial product that enhances Microsoft PowerPoint with constraint-based layout support. In the product, a numeric constraint solver calculates the optimal smart grid layout for a given set of smart elements and placement constraints. A case study verified that the interaction concept as implemented in the prototype increases the efficiency for layout specification by about 20 % over PowerPoint alone, even for expert users.

Acknowledgments

First of all I wish to thank Prof. Dr. Raúl Rojas, since he made this work possible in the first place. I appreciate that Prof. Rojas supported my thesis despite the fact that its nature is not purely academic. Felix von Hundelshausen was my advisor in the work group of Prof. Rojas and I thank him for the good cooperation. I would also like to thank Prof. Dr.-Ing. Klaus-Peter Löhr for his willingness to act as a second reviewer.

Dr.-Ing. Markus Hannebauer and Arno Schödl, PhD, dedicated precious time and constant energy to support my work in all practical respects. Arno was always at hand with helpful technical advice on implementation issues, whereas Markus was readily available whenever I had one of the many questions that arise in the course of writing a diploma thesis. I want to thank both of them and their start-up company think-cell Software GmbH for providing the context and the environment to realize the present work.

The field study and the comparative evaluation were made possible by an unnamed business consultancy office in Berlin, whom I thank for their obliging cooperation and invaluable insights. Particular thanks go to Patrick Weese, a PowerPoint professional who spent many hours giving expert ratings and doing slides under observation.

Annika Hinze provided me with links for better writing skills. (I also borrowed some books from her and must not forget to give them back!) Miriam Busch gave out the LaTeX source code of her own thesis that served me as a working example and made it easy to start off writing. Thank you!

Winfried Schöch, Klaus-Bernd Schürmann, Amber & Till Weinhold and Armin Schöch read my work at different stages. Their valuable comments helped to put this text into shape and the reader has to thank them for an easy and understandable wording (if that is the case).

Last but not least a very special thank-you must be given to my dear wife Rasa and our little daughter(s) for their incredible patience, extraordinary support and unlimited understanding for my occupation throughout the last couple of months.

Contents

List of Figures

1 Introduction

> "*Layout* refers to the process of determining the sizes and positions of visual objects that are part of an information presentation. *Automated layout* refers to the use of a computer program to automate either all or part of the layout process. This field of research lies at the crossroads between artificial intelligence and human computer interaction."
>
> Lok and Feiner in [LF01]

This thesis deals with the human-computer interaction part of a new approach for automated layout. With the notion of *automated layout* I do not refer to graph layout or VLSI layout. These fields of research share some of the terminology used throughout this text, albeit with some subtle differences in meaning. The application domain of my automated layout approach are business slides, which to date are typically designed with Microsoft PowerPoint[1].

A field study in a business consultancy, which is detailed in chapter 2, showed that the tools provided by standard software like PowerPoint are not adequate for efficient slide design. For example, business slide layout typically consists of carefully aligned text boxes and arrows. Since the alignment tools provided by PowerPoint are merely pixel-based and do not create lasting relationships between shapes, setting up a completely aligned layout is a tedious task. In particular, when some more text or some new element needs to be integrated into an existing slide, the entire layout must be updated manually until all requirements are met. Standard requirements are: Certain shapes must have the same size, all text must have the same formatting, all shapes must be horizontally and vertically aligned with some other shapes.

The most regular, grid- or table-like layouts that prevail on typical business slides are apparently suitable for automated layout support. By *automated layout support* I mean the implicit or explicit specification of relations between drawing elements, and the dynamic maintenance thereof during an interactive layout process. A review of the available literature on automated layout is given in chapter 3. It turns out that a number of constraint-based systems for automated layout have been developed in research, but none of them have been successfully applied in practice. This observation is commonly attributed to the fact that those systems are very complex and therefore hard to use. However, the idea of user-specified layout constraints that are interactively maintained when the layout changes, appears to be an appropriate concept for professional slide design.

The start-up company *think-cell*[2] elaborated on that idea and is now developing an add-in for PowerPoint that realizes constraint-based layout automation. They already implemented an architecture, which allows software-driven interaction with

[1] All company and product names used in this text are trademarks or registered trademarks of their respective owners.

[2] My work is based on and part of a product from *think-cell Software GmbH*, Berlin, which is called *think-cell layout*. At the time of writing, it is in early development stage. `http://www.think-cell.com`

PowerPoint – for example, to modify existing shapes or to add a new shape to a slide. Moreover, the people at think-cell developed and implemented a concept which they call the *smart grid*: A collection of horizontal and vertical *smart gridlines* that describes the position of each element on the slide.

Elements that are contained by the smart grid are called *smart elements*, because their locations are no longer pixel-based. Instead, each smart element is bound to a number of smart gridlines and moves along when the gridlines move. The effect being, smart elements that are bound to the same gridline, are aligned – and as long as the binding to the gridline is maintained, they stay aligned even when the over-all slide layout is modified.

At the time of writing, two essential parts of the software were not yet implemented: The numeric constraint solver that computes the "optimal" layout, and the user interface that is necessary to operate the smart grid. It is the theme of this thesis to realize an interaction concept that facilitates an easy and efficient constraint specification for smart elements and their placement in the smart grid. In chapter 4, the smart grid idea is explained in detail and an interaction concept is developed that leads to the specification of a user interface.

While the numeric constraint solver is still not in place, the existing architecture of the think-cell add-in for PowerPoint provided the opportunity to turn my user interface specification into a running prototype. The architecture of the prototype and some of the algorithms used for the implementation are discussed in chapter 5. Finally, chapter 6 evaluates the prototype by means of a comparative case study and quantifies the achievement in terms of the increase of efficiency over plain PowerPoint.

2 Field Study

Jeff Hawkins (PalmPilot inventor)[3]: *"When I design products, I pretend to use them as much as possible before they exist. Meaning, I would put the wood prototype in my shirt pocket and carry it around all the time. If someone phoned, I'd take it out and pretend I was looking up something in the calendar or address book."*

In my case, to acquire a thorough understanding of the problems for which my work attempts to offer a solution, I can do even more than Hawkins with his wooden "prototype". Since the think-cell product is an extension to Microsoft PowerPoint, I could go and watch PowerPoint users in the target setting. A pilot customer of the think-cell company – a mid-sized consultancy office – provided an opportunity for me to do exactly that. The results are presented in this chapter.

2.1 Qualitative Aspects

Before we can even think about a *quantitative* evaluation we need some terminology and some basic understanding of the ongoing processes that involve Microsoft PowerPoint in the context of consultancies. A *qualitative* study provides the domain knowledge or the scale, which can be used to conduct a quantitative evaluation. In order to use time economically, I collected some quantitative data along with the qualitative study. However, the results are presented in distinct sections.

2.1.1 Questions and Methods

In order to design an appropriate user interface, I need to understand the user population, their environment, and their requirements. With regard to the user, the typical level of computer expertise is of special interest. Information about the environment includes typical technical equipment and the working conditions as far as interaction with software is concerned. Furthermore, a contextual task analysis [May99] provides information about the over-all workflow and thereby about the role that the intended software could play and the impact it may have. I used a range of complementary methods to collect and verify the required data.

Observation. While I was present at the consultancy's site, I watched the entire workflow and paid special attention to the people using Microsoft PowerPoint. Doing so, I collected notes which I later aggregated into statements.

Interviews. I asked people to take a few moments and answer those questions which I couldn't answer myself by simply watching – these are typically "why?" questions. If I had a hypothesis in mind, I tried to verify it by asking "right or wrong" kinds of questions. In some cases, I conducted longer and carefully prepared interviews to learn about some part of the workflow which I couldn't

[3] from `http://www.business2.com/articles/mag/0,1640,21,00.html`

cover by observation. An interview leads to in-depth information which cannot easily be generalized as it comes from a single source.

Surveys. Questionnaires are another means to verify hypotheses that came up from observation. As opposed to interviews, surveys support a generalization of a result because input comes from a range of sources. On the other hand, surveys cannot provide very much detail, because the need for statistical evaluation of the responses requires a uniform and restricted format of the answers. A survey can be used to weight the significance of results from observation and interviews.

Hands-on experience. To strengthen the feeling for the work and the problems and to understand the motivation behind some best practices, I tried to use Microsoft PowerPoint myself the way the people at consultancies do. When preparing my own presentations, I used graphical elements and layouts that I had seen before on slides from consultancies. Thus, I developed some skills in using the Microsoft PowerPoint tools for drawing, alignment, z-order, and so forth.

Representativeness of the data. Most observations, interviews and surveys were undertaken in one office of think-cell's pilot customer. In the course of two weeks, I was present at the pilot customer's site on five workdays, which translates to ten 6-hour shifts in the visual pool. The data collected at that site was supplemented and backed up by interviews with consultants from other consultancies. As it turned out, working techniques of different consultancies are extremely similar, thus providing reasonable argument to generalize my findings and conclusions.

2.1.2 User Profiles and Working Environment

There are strictly two types of Microsoft PowerPoint users in a consultancy. On the one hand, there are the *consultants* themselves. On the other hand, there are the visual assistants or simply *visuals*. The entirety of all available visuals at a point in time is referred to as the *pool* or *production*.

Consultants

While the focus and the value of the consultants' work lies in economic improvements of their customers' businesses, creating a convincing presentation as a basis for effective communication with the customer is almost equally high on their priority list.

Consultants do a lot of team work and also split up responsibilities for different aspects of a project and a presentation, respectively. However, the actual creation of slides is done privately. The consultant's typical workplace is constantly moving between the home office and one or more customer sites. She[4] has a laptop computer

[4] For ease of writing, I always refer to "the user" as *she* with no gender-specific meaning implied.